CMPG-767 Digital Image Processing

FREQUENCY DOMAIN PROCESSING
FOURIER TRANSFORM

Frequency Domain

- Frequency domain is the domain for analysis of signals with respect to frequency
- Time domain representation of a signal (in 1-D case) and spatial domain (in 2-D case) shows how the signal changes in time or space, respectively
- Frequency domain representation shows how much of the signal lies within each given frequency band over a range of frequencies

Importance of Frequency Domain

- Frequency domain is very important in signal processing
- It shows which frequencies participate in the formation of a signal and shows their contribution to this formation, as well as the phase shift for each frequency
- It is very important to be able to design filters that eliminate (correct) contribution of some certain frequencies to a signal, passing other frequencies with no changes
- It is also very important to be able to restore a signal from distortions caused by its convolution with a distorting function (typical example – blur in image processing)
- It can also be used for solving image recognition problems, for example, for object localization

Frequency Domain Representation

- A mechanism, which is used to get the frequency domain representation of a signal defined in the time or spatial domain is the Fourier transform

- The inverse Fourier transform is used to get the time or spatial domain representation of a signal from its frequency domain representation

Jean Baptiste Joseph Fourier (1768-1830)

- French mathematician and physicist
- Investigating how it is possible to compute different functions representing them through some well-known functions, Fourier found (**1807**, published in **1822**) that any periodic function can be represented as the sum of sins and/or cosines of different frequencies (Fourier series)
- Non-periodic, but bounded functions, can be represented as the integral of sins and/or cosines multiplied by a weighting function (Fourier transform)

Fourier Representation

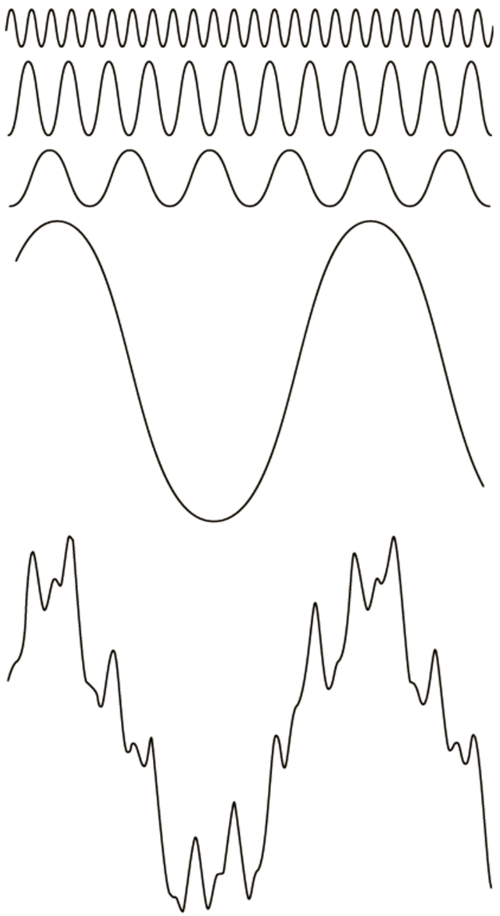


FIGURE 4.1 The function at the bottom is the sum of the four functions above it. Fourier's idea in 1807 that periodic functions could be represented as a weighted sum of sines and cosines was met with skepticism.

Orthogonal Functions

The continuous functions of a real variable $\{u_j(t)\} = \{u_0(t), u_1(t), \ldots\}$ are called **orthogonal** on the interval $(t_0, t_0 + T)$, if

$$\int_{t_0}^{t_0+t} u_j(t) u_k(t) dt = \begin{cases} c, & \text{if } j = k \\ 0, & \text{if } j \neq k \end{cases}$$

If $c=1$ then then functions $\{u_i(t)\} = \{u_0(t), u_1(t), \ldots\}$

are called **orthonormal**

Orthogonal Vectors

A set of $\boldsymbol{n}$-dimensional vectors $\left\{U_j = (u_1^j, \ldots, u_n^j)\right\} = \left\{U_0, U_1(t), \ldots\right\};\ u_s^j \in \mathbb{C}$ over the field of complex numbers is called **orthogonal**, if

$$\underbrace{\left(U_j, U_k\right)}_{\text{dot product}} = u_1^j \overline{u}_1^k + \ldots + u_n^j \overline{u}_n^k = \begin{cases} c, & \text{if } j = k \\ 0, & \text{if } j \neq k \end{cases}$$

If $u = a + bi$ then $\overline{u} = a - bi$ is complex-conjugated to u

If $c=1$ then vectors $\left\{U_j = (u_1^j, \ldots, u_n^j)\right\} = \left\{U_0, U_1, \ldots\right\};\ u_s^j \in \mathbb{C}$

are called **orthonormal**

Signals and Functions

- Any signal can be treated as a function, which is defined either on the temporal interval (for 1-D signals) or on the space interval (for 2-D signals)
- For example, $x(t)$ is a 1-D signal, which is defined on the interval $\left(t_0, t_0 + T\right)$
- $f\left(x, y\right)$ is a 2-D signal. Which is defined on the interval

$$\left((x_0, x_0 + N), (y_0, y_0 + M)\right)$$

Signals and Functions

- Any signal can be represented as the following orthogonal series (can be broken down in the combination of orthogonal functions)

$$x(t) = \sum_{j=0}^{\infty} a_j u_j(t)$$

- To find coefficients a_j it is enough to multiply the last equation by $u_k(t)$ and to integrate it:

$$\int_{t_0}^{t_0+T} x(t) u_k(t) dt = \int_{t_0}^{t_0+T} \sum_{j=0}^{\infty} a_j u_j(t) u_k(t) dt$$

Signals and Functions

- Taking into account that functions $\{u_i(t)\} = \{u_0(t), u_1(t), \ldots\}$ are orthogonal, we obtain

$$\int_{t_0}^{t_0+T} x(t)u_k(t)dt = \int_{t_0}^{t_0+T} \sum_{j=0}^{\infty} a_j \underbrace{u_j(t)u_k(t)}_{=c, \text{if } j=k} dt \Rightarrow$$

$$\Rightarrow a_k = \frac{1}{c} \int_{t_0}^{t_0+T} x(t)u_k(t)dt \;;\; k = 0,1,2,\ldots$$

Fourier Representation of Signals

- Let a set $\{u_j(t)\} = \{u_0(t), u_1(t), \ldots\}$ consists of functions

$$\{1,\ \cos j\omega_0 t,\ \sin j\omega_0 t\},\ j = 1, 2, \ldots$$

- Then $x(t) = \sum_{j=0}^{\infty} a_j u_j(t) =$

$$= a_0 + \sum_{j=1}^{\infty} b_j \cos j\omega_0(t) + \sum_{j=1}^{\infty} c_j \sin j\omega_0(t)$$

- where ω_0 is called a main angular frequency, which determines a period $T = 2\pi / \omega_0$ of $\cos j\omega_0 t,\ \sin j\omega_0 t$
- If k is a frequency , $\omega = 2\pi k$ is an angular frequency

Fourier Representation of Signals

- Taking into account that (***i*** is an imaginary unit)

$$\cos j\omega_0 t = \frac{1}{2}\left(e^{ij\omega_0 t} + e^{-ij\omega_0 t}\right)$$

$$\sin j\omega_0 t = \frac{1}{2i}\left(e^{ij\omega_0 t} - e^{-ij\omega_0 t}\right)$$

- then

$$x(t) = \sum_{j=0}^{\infty} d_j e^{ij\omega_0 t} = \sum_{j=0}^{\infty} d_j e^{ij\frac{2\pi}{T}t}$$

Fourier Representation of Signals

- Coefficients d_j can be obtained from

$$x(t) = \sum_{j=0}^{\infty} d_j e^{ik\omega_0 t} \quad \text{as follows:}$$

$$d_j = \int_{-T/2}^{T/2} x(t) e^{-ik\omega_0 t} dt, \quad \omega_0 = \frac{2\pi}{T} = 2\pi k$$

Fourier Transform

- Thus, the Fourier transform of $x(t)$ is

$$F_x(\omega) = \int_{-\infty}^{\infty} x(t)e^{-i\omega t}dt = F_x(k) = \frac{1}{T}\int_{-\infty}^{\infty} x(t)e^{-i2\pi kt}dt$$

- where $$\omega = k\omega_0 = k\underbrace{\left(2\pi / T\right)}_{\omega_0} = k2\pi / T, k = 0,1,2,\ldots$$

  is a continuous angular frequency

- and k is a frequency

Inverse Fourier Transform

- The signal $x(t)$ can be restored from its Fourier transform $F_x(\omega)$ using **inverse** Fourier transform by

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F_x(\omega)e^{i\omega t}d\omega = \frac{1}{2\pi}\int_{-\infty}^{\infty} F_x(j)e^{i2\pi kt}du$$

Discrete Fourier Transform

- If the discrete signal $x(t)$ contains exactly N samples (this means that $x(t)$ has been obtained by sampling of the continuous signal during N equal time intervals), then its Discrete Fourier Transform is

$$F_x(k) = \frac{1}{N}\sum_{j=0}^{N-1} x(j)e^{-i2\pi jk/N},\ k = 0,1,\ldots,N-1$$

Inverse Discrete Fourier Transform

- The signal $x(t)$ can be restored from its discrete Fourier transform using **inverse** Fourier transform as follows:

$$x(j) = \sum_{k=0}^{N-1} F_x(k) e^{i2\pi kj/N}, \; j = 0, 1, \ldots, N-1$$

Discrete Fourier Transform

- Basic functions of the Discrete Fourier Transform (*discrete exponential functions*)

$$f_k = e^{i2\pi jk/N},\ j = 0,1,\dots,N-1;\ k = 0,1,\dots,N-1$$

form a complete system of the orthogonal functions, which means that

$$\underbrace{(f_s, f_r)}_{\text{dot product}} = \begin{cases} N, & \text{if } s = r \\ 0, & \text{if } s \neq r \end{cases}$$

$$\forall s,r \in \{0,1,\dots,N-1\}\ \exists k \in \{0,1,\dots,N-1\} : f_s \circ f_r = f_k$$

where $^{\circ}$ is a component-wise multiplication of vectors

Power and Phase Spectrum

- The absolute values of the Fourier transform coefficients of discrete signal $x(t)$ form a Power Spectrum (or simply a spectrum, or magnitude)

$$\left|F_x(k)\right| = \left|\frac{1}{N}\sum_{j=0}^{N-1} x(j)e^{-i2\pi jk/N}\right|, \; k = 0,1,\dots,N-1$$

- while the arguments of the of the Fourier transform coefficients of discrete signal $x(t)$ form a Phase Spectrum (phase shifts)

$$\varphi_x(k) = \arg F_x(k) = \arg \frac{1}{N}\sum_{j=0}^{N-1} x(j)e^{-i2\pi jk/N}, \; k = 0,1,\dots,N-1$$

Power and Phase Spectrum

- The Fourier transform can be written also as

$$F_x(k) = \left|F_x(k)\right| e^{i \arg F_x(k)},\ k = 0,1,\ldots,N-1$$

- If a signal is reconstructed from the Fourier transform, then

$$x(j) = \sum_{k=0}^{N-1} \left|F_x(k)\right| e^{i \arg F_x(k)} e^{i 2\pi kj/N} =$$

$$= \sum_{k=0}^{N-1} \left|F_x(k)\right| e^{i\left(2\pi kj/N + \arg F_x(k)\right)},\ j = 0,1,\ldots,N-1$$

- $\arg F_x(k)$ is the phase shift

Phase vs. Magnitude

- Oppenheim, A.V.; Lim, J.S., **The importance of phase in signals**, IEEE Proceedings, v. 69, No 5, 1981, pp.: 529- 541

- In this paper, it was shown that the **phase** in the Fourier spectrum of a signal is much more informative than the **magnitude**: particularly in the Fourier spectrum of images phase contains the information about all shapes, edges, orientation of all objects, etc.

Importance of Phase

- Phase contains the information about all the edges, shapes, and their orientation

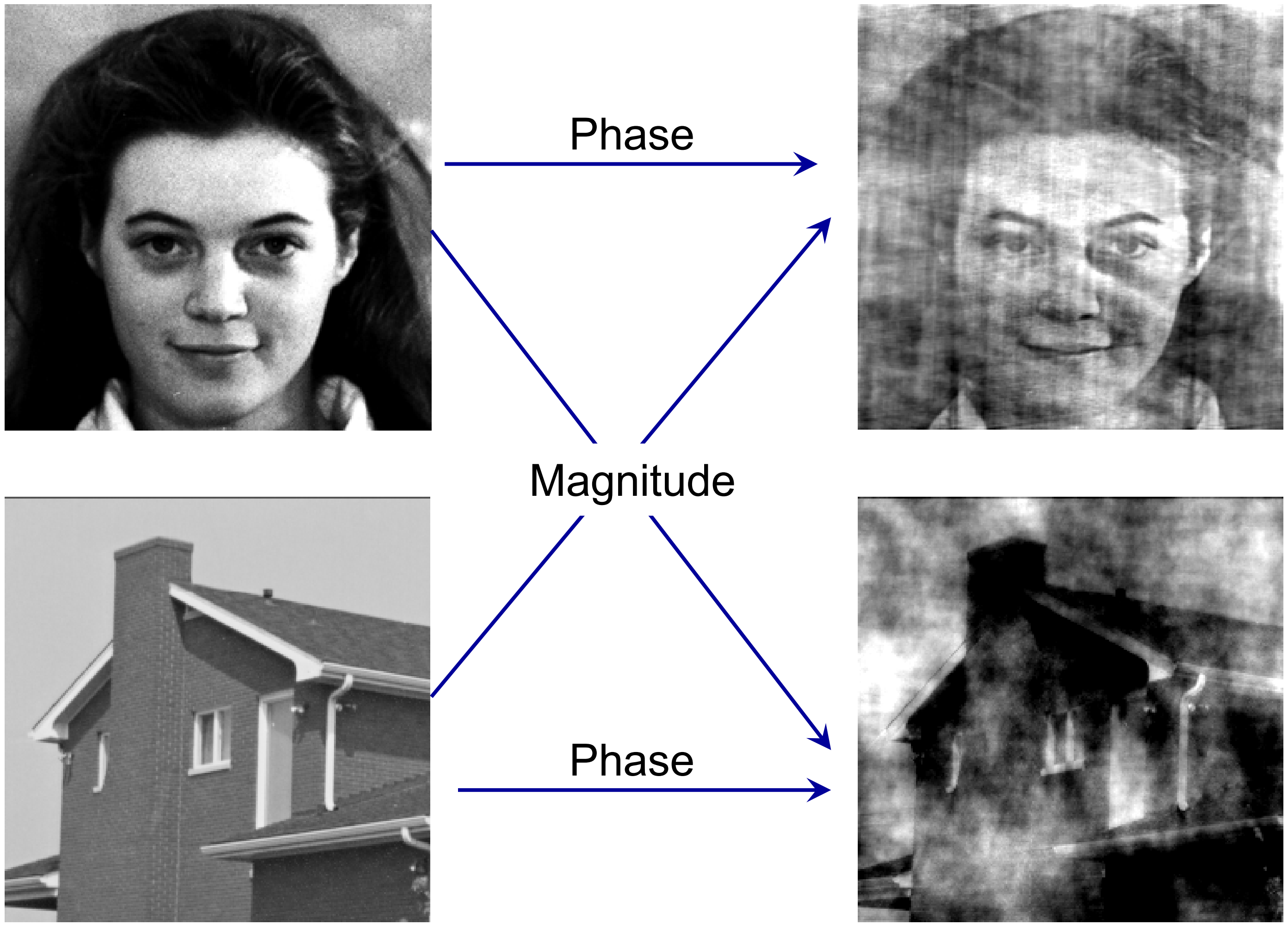
This image results from the inverse Fourier transform applied to the spectrum synthesized from the original phases, but with magnitudes substituted with the constant “1”

Image Recognition: Features Selection

- Thus, the Fourier Phase Spectrum can be a very good source of features that describe all objects represented by the corresponding signals.

- The Power Spectrum (magnitude) describes some global signal properties (for, example blur, noise, cleanness, contrast, brightness, etc. for images).

Importance of Phase



Phase and Magnitude

Magnitude contains the information about the signal's properties

(a)

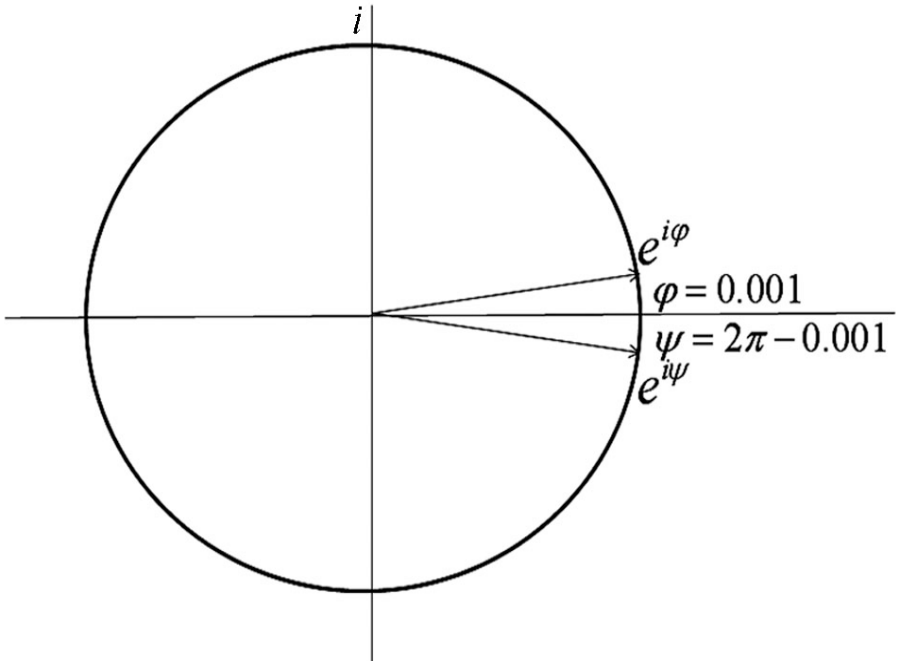
(b)

Phase (a) & **Magnitude** (b)

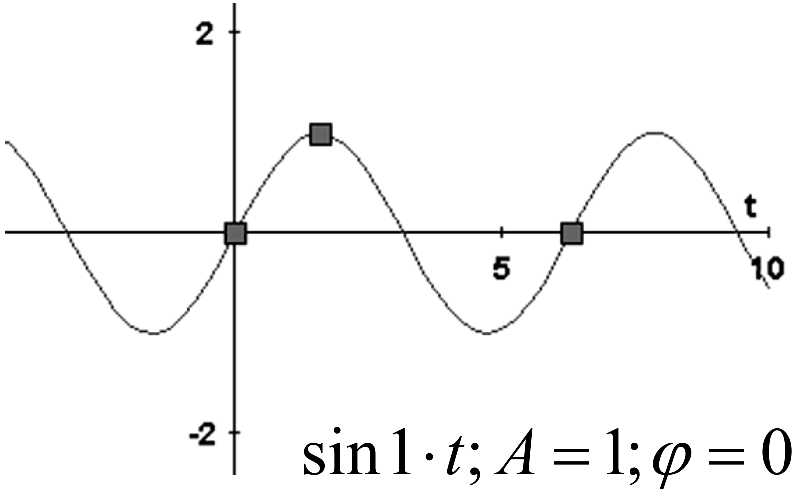
Phase (b) & **Magnitude** (a)

Proper treatment of phase

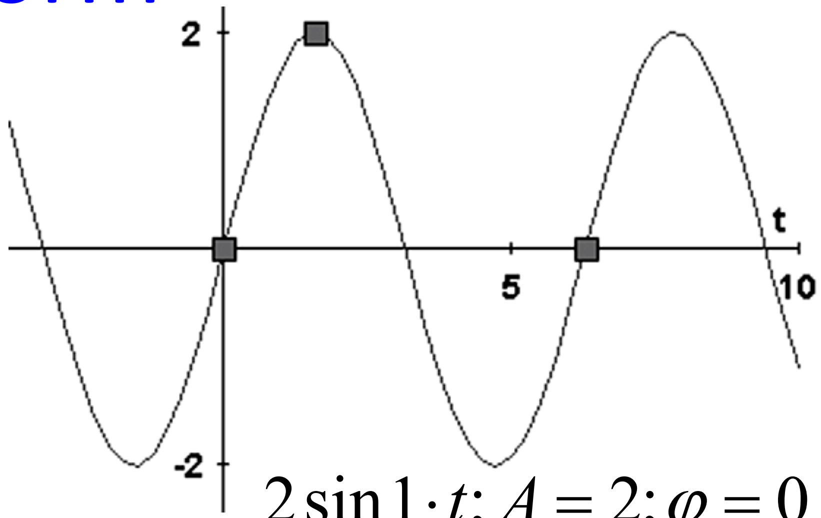
- If phases are treated as some abstract real numbers, we may treat phases such as $\varphi = 0.001$ and $\psi = 2\pi - 0.001 = 6.282$ as much different values. This would be incorrect because in fact they are very close to each other



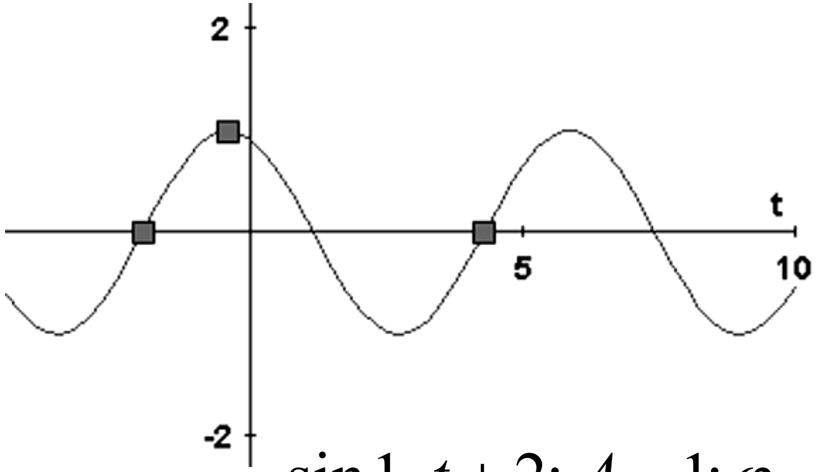
A role of phase and magnitude in the Fourier transform



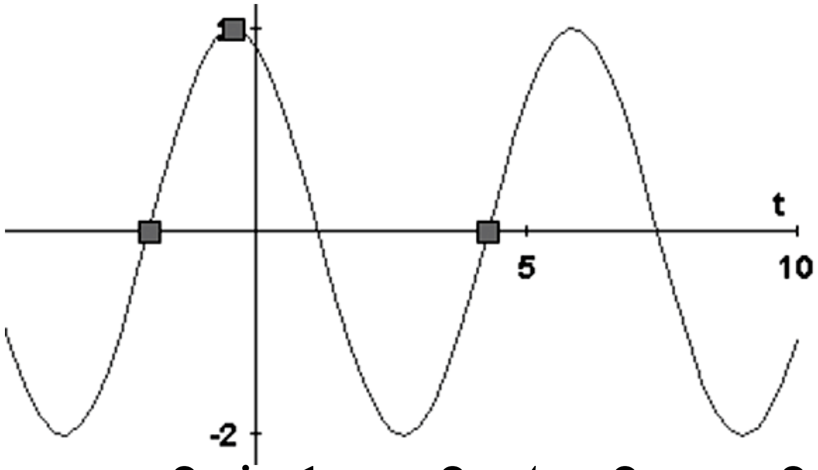
$\sin 1 \cdot t; A = 1; \varphi = 0$



$2\sin 1 \cdot t; A = 2; \varphi = 0$



$\sin 1 \cdot t + 2; A = 1; \varphi = 2$



$2\sin 1 \cdot t + 2; A = 2; \varphi = 2$

Convolution

- Convolution of two continuous functions $f(t)$ and $w(t)$ is a process of flipping (rotating by 180^0) one function about its origin and sliding it past the other

$$g(t) = f(t) * w(t) = \int_{-\infty}^{\infty} w(\tau) f(t \div \tau) d\tau$$

where $\div$ stands for the flipping, τ is the displacement needed to slide one function past the other

Discrete Convolution

- Convolution of two discrete functions $f(t)$ and $w(t)$ is also a process of flipping (rotating by 180^0) one function about its origin and sliding it past the other

$$g(t) = f(t) * w(t) = \sum_{k=0}^{N-1} w(k) f(t \div k); t = 0,1,\ldots,N-1$$

$$\boxed{(t+k) \bmod N}$$

where $\div$ stands for the flipping (shift corresponding to the basis, where the convolution is taken; for the Fourier basis this is the circular shift), k is the displacement needed to slide one function past the other

Circular Shift

$f(t)$	A	B	C	D	E	F	G

Circular shift by 3:

$f(t \div 3)$	E	F	G	A	B	C	D

Convolution Theorem

- Theorem. Fourier transform of the convolution of signals $x(t)$ and $y(t)$ equals to the product of the Fourier transforms of these signals

$$F_{x*y}(k) = F_x(k) \cdot F_y(k); k = 0,1,...,N-1$$

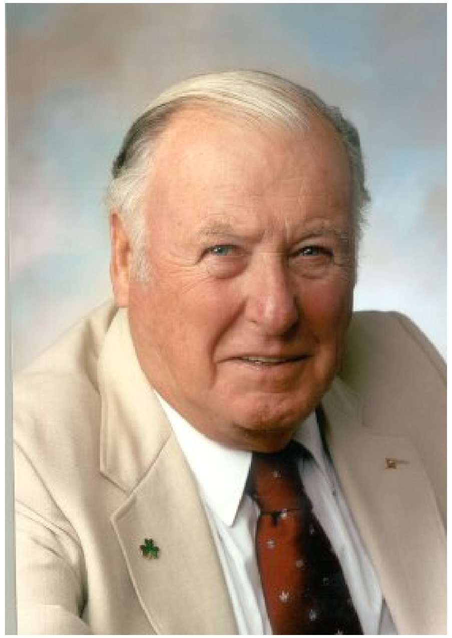
- Corollary. The convolution of signals $x(t)$ and $y(t)$ equals the inverse Fourier transform of the product of their Fourier transforms

$$g(j) = x(j) * y(j) = F^{-1}_{F_x \cdot F_y}\left(j\right); j = 0,1,...,N-1$$

Fast Fourier Transform (FFT)

- Fast Fourier Transform is a highly efficient algorithm for computation of the discrete Fourier transform of 1-D signal of the length N equal to any power of 2.
- Its efficiency is $N \log_2 N$ while the efficiency of a straightforward algorithm is N^2
- This algorithm was invented by James Cooley and John Tukey in 1965 and commonly known as FFT or Cooley-Tukey algorithm

Fast Fourier Transform (FFT)



James Cooley, 1926-2016
received a B.A. degree in Math in 1949
from **Manhattan University**

John Tukey 1915-2000

Inventors of the **Fast Fourier Transform** algorithm commonly known as **FFT**

J. W. Cooley and J. W. Tukey, An algorithm for the machine calculation of complex Fourier series.
Mathematics of Computation, vol. 19, pp. 297-301, **1965**

2-D Continuous Fourier Transform

- Continuous Fourier transform of the function $g(x, y)$ of two continuous variables

$$F_g(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y)e^{-i2\pi(xu+yv)}dxdy$$

- Continuous inverse 2-D Fourier transform

$$g(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F_g(u,v)e^{i2\pi(xu+yv)}dudv$$

2-D Discrete Fourier Transform

- Discrete Fourier transform of the digital image $g(x, y)$ of size $M \text{ x } N$

$$F_g(u,v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} g(x,y)e^{-i2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

- Discrete inverse 2-D Fourier transform

$$g(x,y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F_g(u,v)e^{i2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

Computation of the 2-D Discrete Fourier Transform

- As $e^{-i2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)} = e^{-i2\pi\frac{ux}{M}} \cdot e^{-i2\pi\frac{vy}{N}}$, the 2-D DFT can be **separated** – it can be found first with respect to one dimension (for rows) and then with respect to another one (for columns):

$$F_g(u,v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} g(x,y)e^{-i2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)} = \frac{1}{M}\sum_{x=0}^{M-1}\left(\frac{1}{N}\sum_{y=0}^{N-1} g(x,y)e^{-i2\pi\frac{vy}{N}}\right)e^{-i2\pi\frac{ux}{M}}$$

Computation of the 2-D Discrete Fourier Transform

- In practical implementation, 2-D DFT of a digital image should be calculated as follows:

1) Fourier transform of the rows should be taken, row by row and the results should be stored in a matrix (2-D array)
2) This matrix should be transposed
3) Then Fourier transform of the rows of this transposed matrix should be taken and the results should be stored in a matrix (2-D array), which contains the 2-D Fourier transform of the initial image

Calculation of the 2-D Discrete Fourier Transform

- Considering that the Fast Fourier Transform algorithm exists only for the dimensions that equal to some powers of two, an image whose sizes are not powers of two, should be extended to the next closest power of two

- This extension should be done better by mirroring in all directions rather than by zero-padding, to avoid "boundary effects"

Important Properties of 2-D Discrete Fourier Transform

	Spatial Domain†		Frequency Domain†
1)	$f(x, y)$ real	$\Leftrightarrow$	$F^*(u, v) = F(-u, -v)$
2)	$f(x, y)$ imaginary	$\Leftrightarrow$	$F^*(-u, -v) = -F(u, v)$
3)	$f(x, y)$ real	$\Leftrightarrow$	$R(u, v)$ even; $I(u, v)$ odd
4)	$f(x, y)$ imaginary	$\Leftrightarrow$	$R(u, v)$ odd; $I(u, v)$ even
5)	$f(-x, -y)$ real	$\Leftrightarrow$	$F^*(u, v)$ complex
6)	$f(-x, -y)$ complex	$\Leftrightarrow$	$F(-u, -v)$ complex
7)	$f^*(x, y)$ complex	$\Leftrightarrow$	$F^*(-u - v)$ complex
8)	$f(x, y)$ real and even	$\Leftrightarrow$	$F(u, v)$ real and even
9)	$f(x, y)$ real and odd	$\Leftrightarrow$	$F(u, v)$ imaginary and odd
10)	$f(x, y)$ imaginary and even	$\Leftrightarrow$	$F(u, v)$ imaginary and even
11)	$f(x, y)$ imaginary and odd	$\Leftrightarrow$	$F(u, v)$ real and odd
12)	$f(x, y)$ complex and even	$\Leftrightarrow$	$F(u, v)$ complex and even
13)	$f(x, y)$ complex and odd	$\Leftrightarrow$	$F(u, v)$ complex and odd

†Recall that x, y, u, and v are *discrete* (integer) variables, with x and u in the range $[0, M - 1]$, and y, and v in the range $[0, N - 1]$. To say that a complex function is *even* means that its real *and* imaginary parts are even, and similarly for an odd complex function.

TABLE 4.1 Some symmetry properties of the 2-D DFT and its inverse. $R(u, v)$ and $I(u, v)$ are the real and imaginary parts of $F(u, v)$, respectively. The term *complex* indicates that a function has nonzero real and imaginary parts.

Discrete Cosine Transform (DCT)

- Discrete Cosine Transform of signal $x(t)$ is a shifted Fourier transform determined by

$$C_x(k) = \frac{1}{N}\sum_{j=0}^{N-1} x(j)\cos\left[\frac{\pi}{N}\left(j+\frac{1}{2}\right)k\right],\ k = 0,1,\ldots,N-1$$

- Signal $x(t)$ can be restored from its discrete Cosine transform as follows:

$$x(j) = \frac{1}{2}C_x(0)\sum_{k=1}^{N-1} C_x(k)\cos\left[\frac{\pi}{N}\left(j+\frac{1}{2}\right)k\right],\ j = 0,1,\ldots,N-1$$

Discrete Cosine Transform (DCT)

- Discrete Cosine Transform, being a shifted Fourier transform, expresses a finite sequence of data points in terms of a sum of cosine functions oscillating at different frequencies

- DCT is equivalent to DFT of roughly twice the length, operating on real data with even symmetry (since the Fourier transform of a real and even function is real and even), where the input and/or output data are shifted by half a sample.

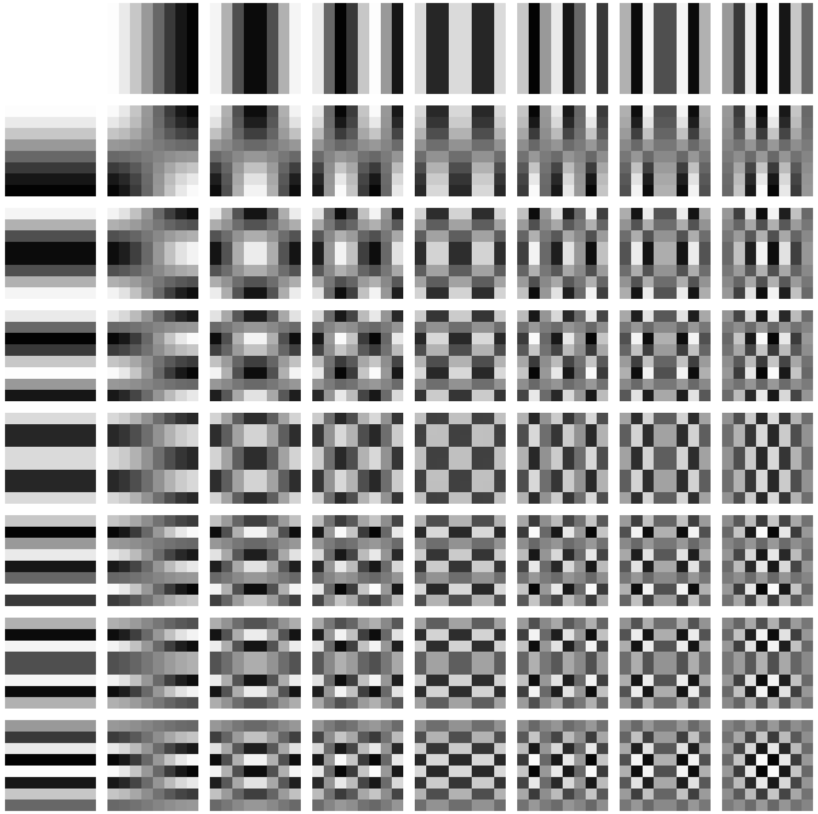
Discrete Cosine Transform (DCT)

- Discrete Cosine Transform, being a real-valued, but holding basically the same properties as Discrete Fourier Transform is used in many applications, particularly in lossy data compression (jpeg for image data compression and mp3 for audio data compression)
- Lossy frequency domain compression is based on the possibility of neglecting or less accurate encoding of the transform coefficients in the high frequency area

2D Discrete Cosine Transform

- 2D DCT is a separable transform (as a 2D DFT) and should be calculated in the same way

- 8x8 DCT of an image fragment with respect to 2D frequencies (8x8 DCT is used in jpeg compression)



Sampling

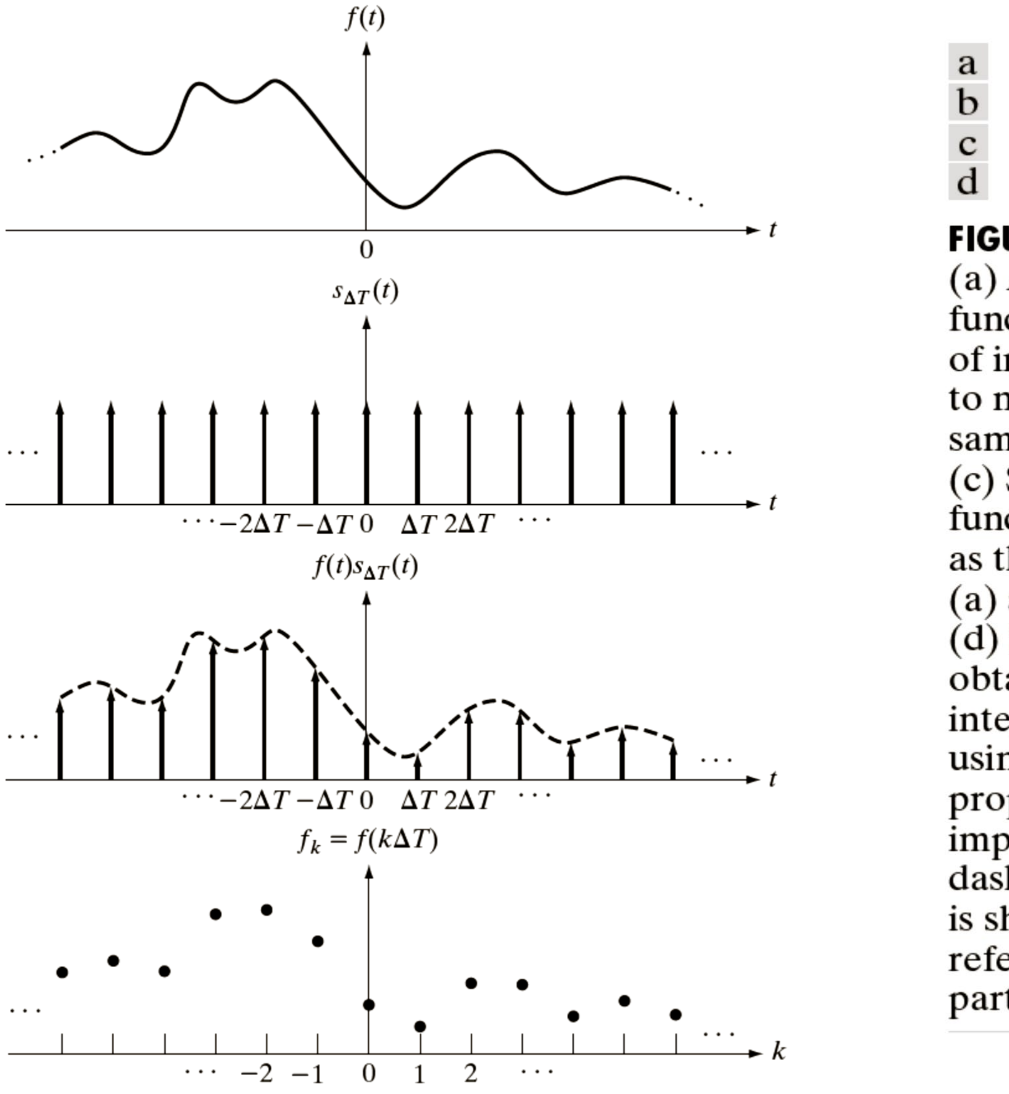
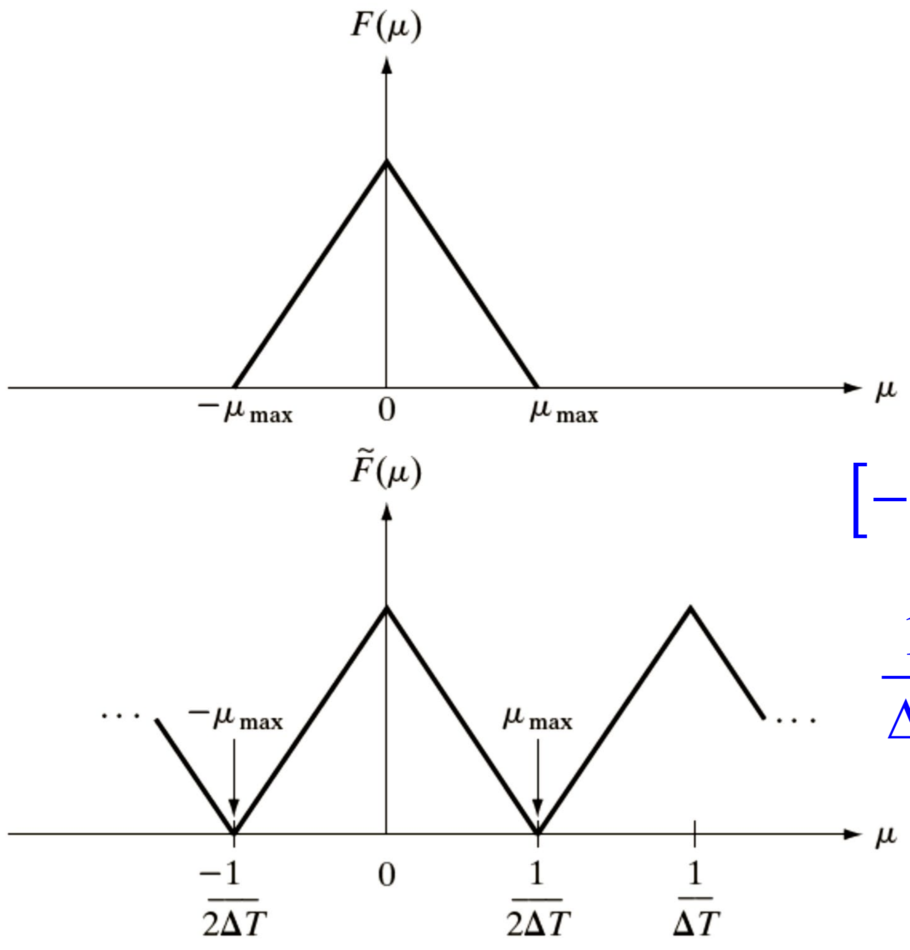


FIGURE 4.5 (a) A continuous function. (b) Train of impulses used to model the sampling process. (c) Sampled function formed as the product of (a) and (b). (d) Sample values obtained by integration and using the sifting property of the impulse. (The dashed line in (c) is shown for reference. It is not part of the data.)

Sampling



a
b

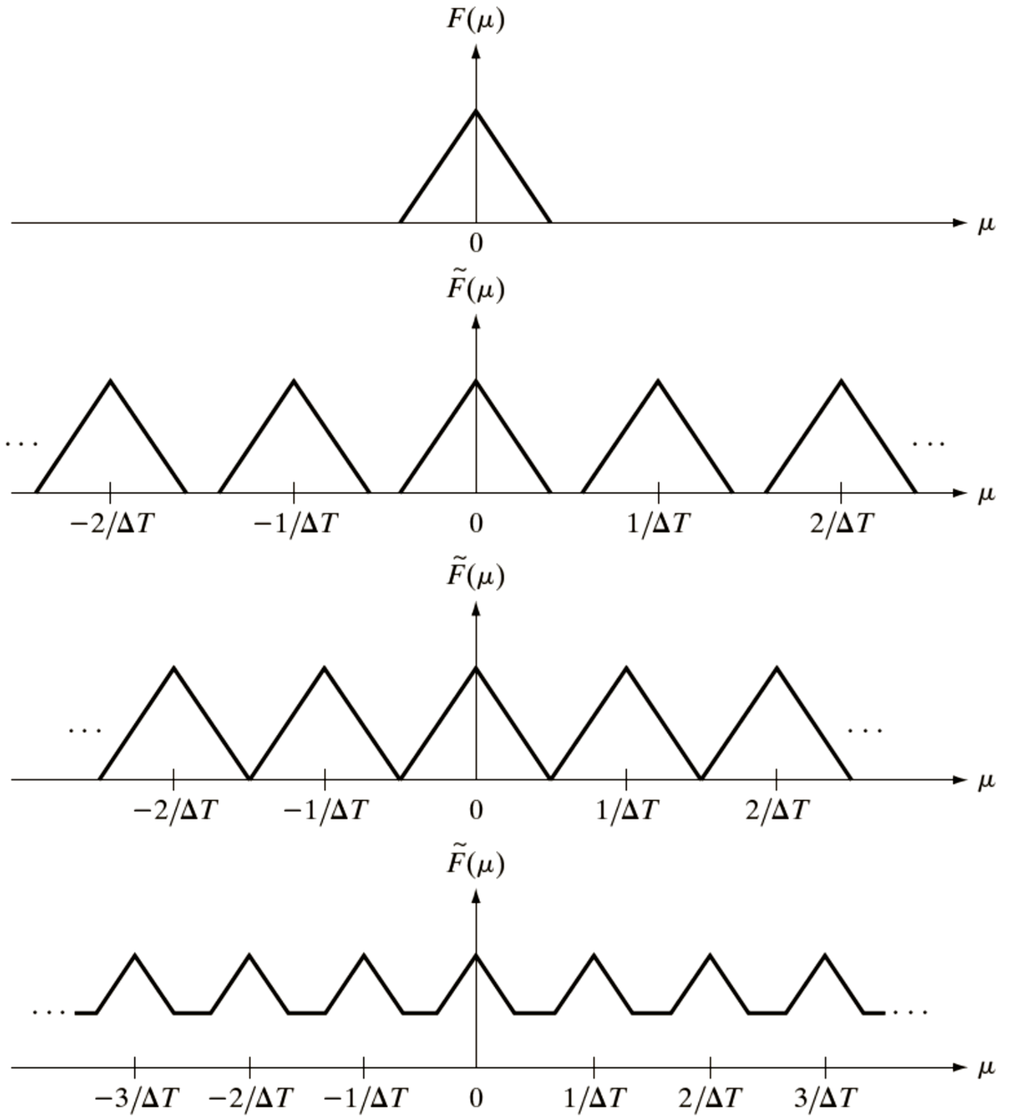
FIGURE 4.7
(a) Transform of a band-limited function.
(b) Transform resulting from critically sampling the same function.

$\left[-\mu_{max}, \mu_{max}\right]$ - The frequency range of a band-limited signal

$\frac{1}{\Delta t} = 2\mu_{max}$ - The Nyquist rate

To recover a signal from its sampled representation, the sampling rate must exceed the Nyquist rate: $\frac{1}{\Delta t} > 2\mu_{max}$

Sampling



Oversampling

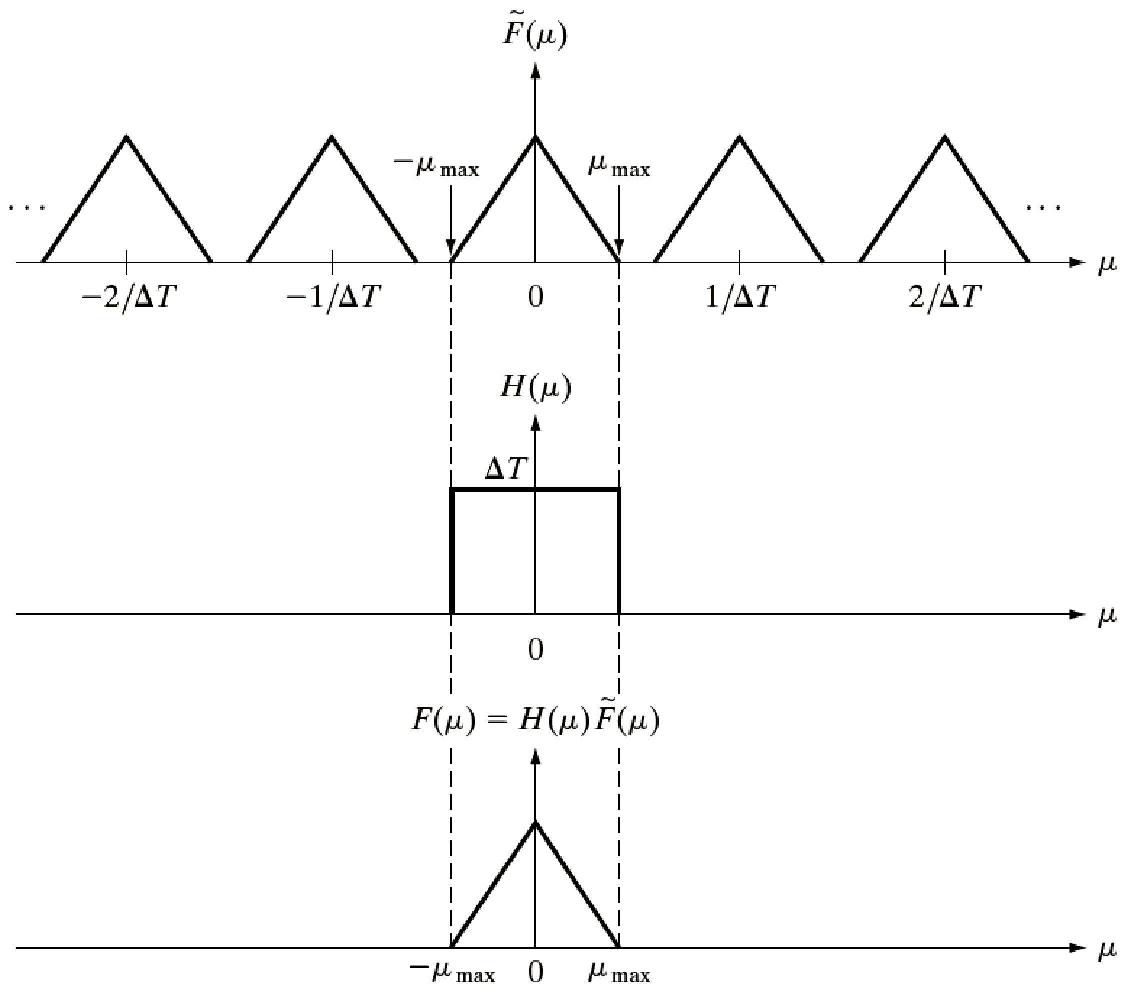
Critical sampling

Undersampling

a
b
c
d

FIGURE 4.6 (a) Fourier transform of a band-limited function. (b)–(d) Transforms of the corresponding sampled function under the conditions of over-sampling, critically-sampling, and under-sampling, respectively.

Sampling – Reconstruction from the Oversampled Discrete Signal



a
b
c

FIGURE 4.8 Extracting one period of the transform of a band-limited function using an ideal lowpass filter.

Aliasing

- Aliasing (frequency aliasing) is a process in which high frequency components of a continuous function "masquerade" as lower frequencies in the sampled function ("alias" means a false identity)
- Spatial aliasing is due to undersampling
- Temporal aliasing is related to time intervals between sequence of images (e.g. rotation of wheels backwards in a movie, etc.)

Undersampling and Aliasing

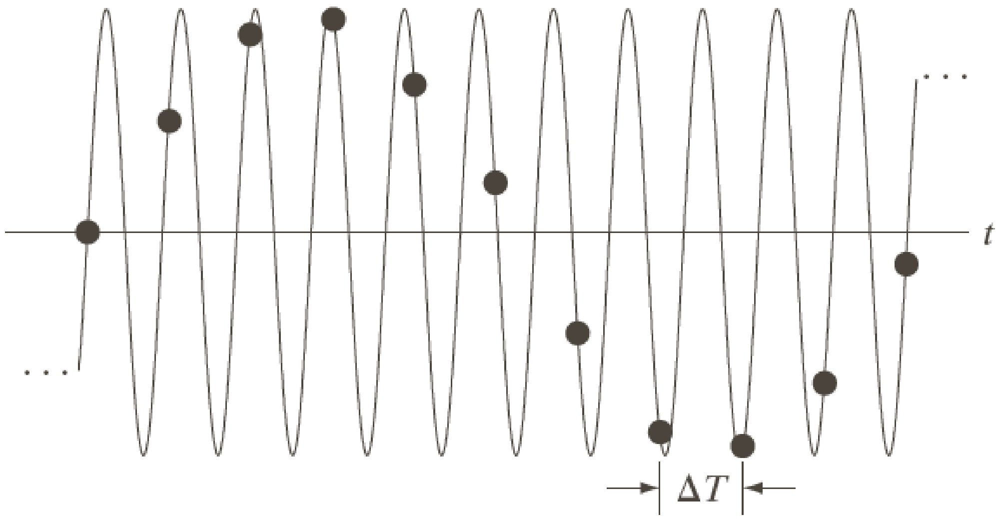


FIGURE 4.10 Illustration of aliasing. The under-sampled function (black dots) looks like a sine wave having a frequency much lower than the frequency of the continuous signal. The period of the sine wave is 2 s, so the zero crossings of the horizontal axis occur every second. ΔT is the separation between samples.

Undersampling and Aliasing

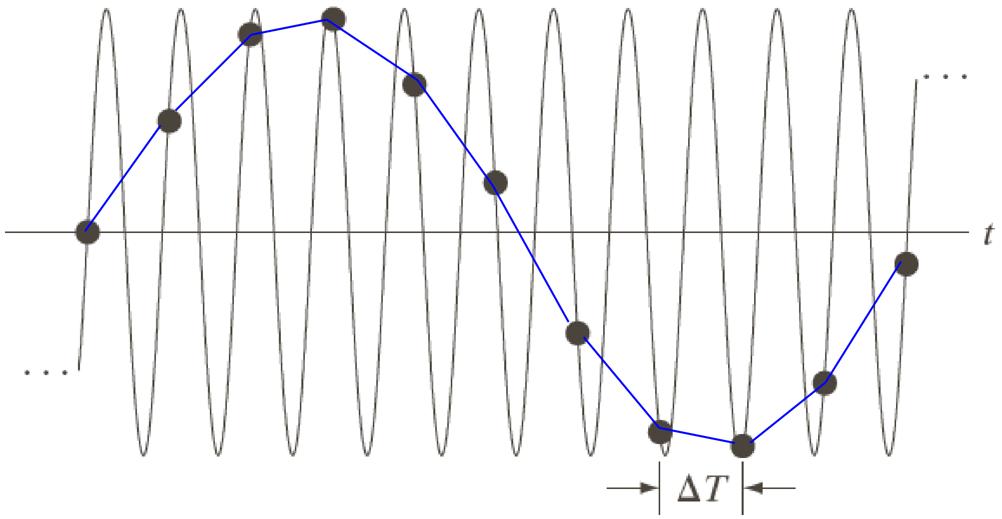
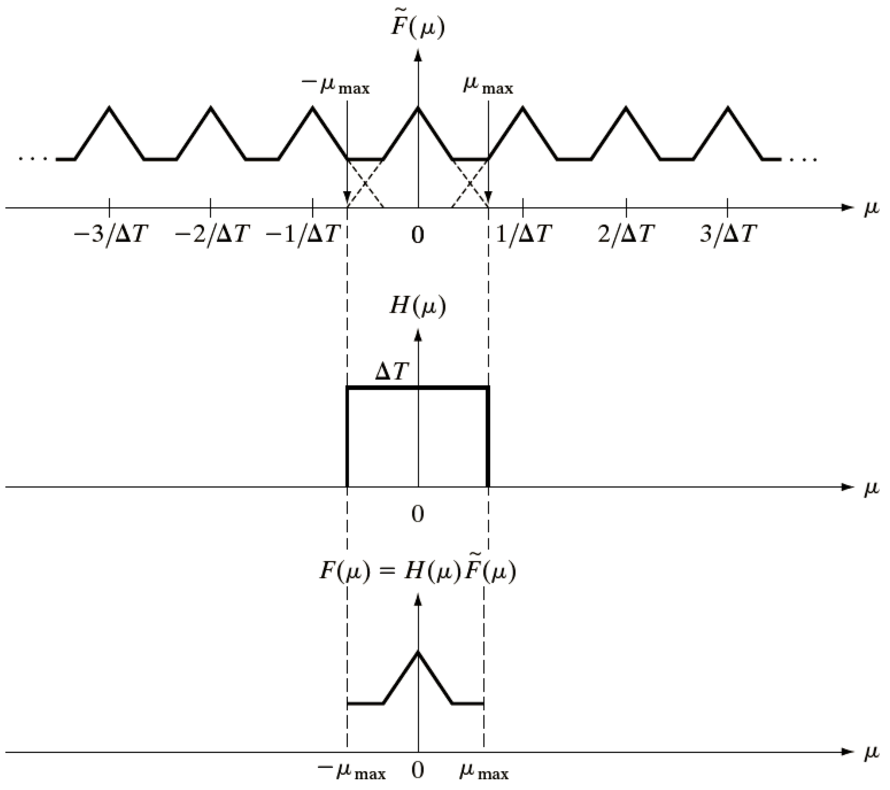


FIGURE 4.10 Illustration of aliasing. The under-sampled function (black dots) looks like a sine wave having a frequency much lower than the frequency of the continuous signal. The period of the sine wave is 2 s, so the zero crossings of the horizontal axis occur every second. ΔT is the separation between samples.

Sampling – Reconstruction from the Undersampled Discrete Signal



a
b
c

FIGURE 4.9 (a) Fourier transform of an under-sampled, band-limited function. (Interference from adjacent periods is shown dashed in this figure). (b) The same ideal lowpass filter used in Fig. 4.8(b). (c) The product of (a) and (b). The interference from adjacent periods results in aliasing that prevents perfect recovery of $F(\mu)$ and, therefore, of the original, band-limited continuous function. Compare with Fig. 4.8.

- **Lowering** the sampling rate below the Nyquist rate leads to **impossibility to isolate a single period of the Fourier transform**